前端資料收集工具簡介

台灣大學 農藝系
胡凱康 王裕文

一、引言

種原庫的建立，從育種者的角度來看，它是一個提供育種材料的集中保管場所，因為所收納的各種種原材料，將因為有專業與專職的人員進行包括如定期的發芽率(以種子形式保存的材料)監測與適時的種原材料更新，使各材料的遺傳特性得以保持，更提供一個種原材料交換的通道，使所收集的種原得以被充分的開發利用。

種原庫除了提供實質的材料保存與交換功能以外，也是一個遺傳資源變異累積的資料庫，因為各個種原材料在提交到種原庫時，各材料提供者必須提供所指定的種原特性資料，其後各材料的索取者與使用者也有義務在取得材料進行觀察試驗後，提報相關的特性資料。透過這些特性資料的累積，一個資料庫就已經成型了。

現階段的種原庫利用現況，使用者通常在索取材料之前，可以利用電腦指定特定的性狀，查詢種原資料庫內所蒐集保存的相關材料。目前使用者主要以敘述性的性狀文字描述來查詢資料庫，但受限於文字敘述的限制，取得的資料大多只能提供約略的參考，使用者多半需要自行繁殖材料，實地進行性狀觀察來作為最終選擇的依據。為了取得最終所需的材料，避免遺珠之憾，使用者可能需要增加索取的種原數目，如此，一方面將造成使用者取得材料後額外的評選負擔，一方面也增加種原庫維護與種原更新的負擔。因此，如果能夠提供更精確的種原特性資料，使用者就不需要擴大指定性狀的範圍。

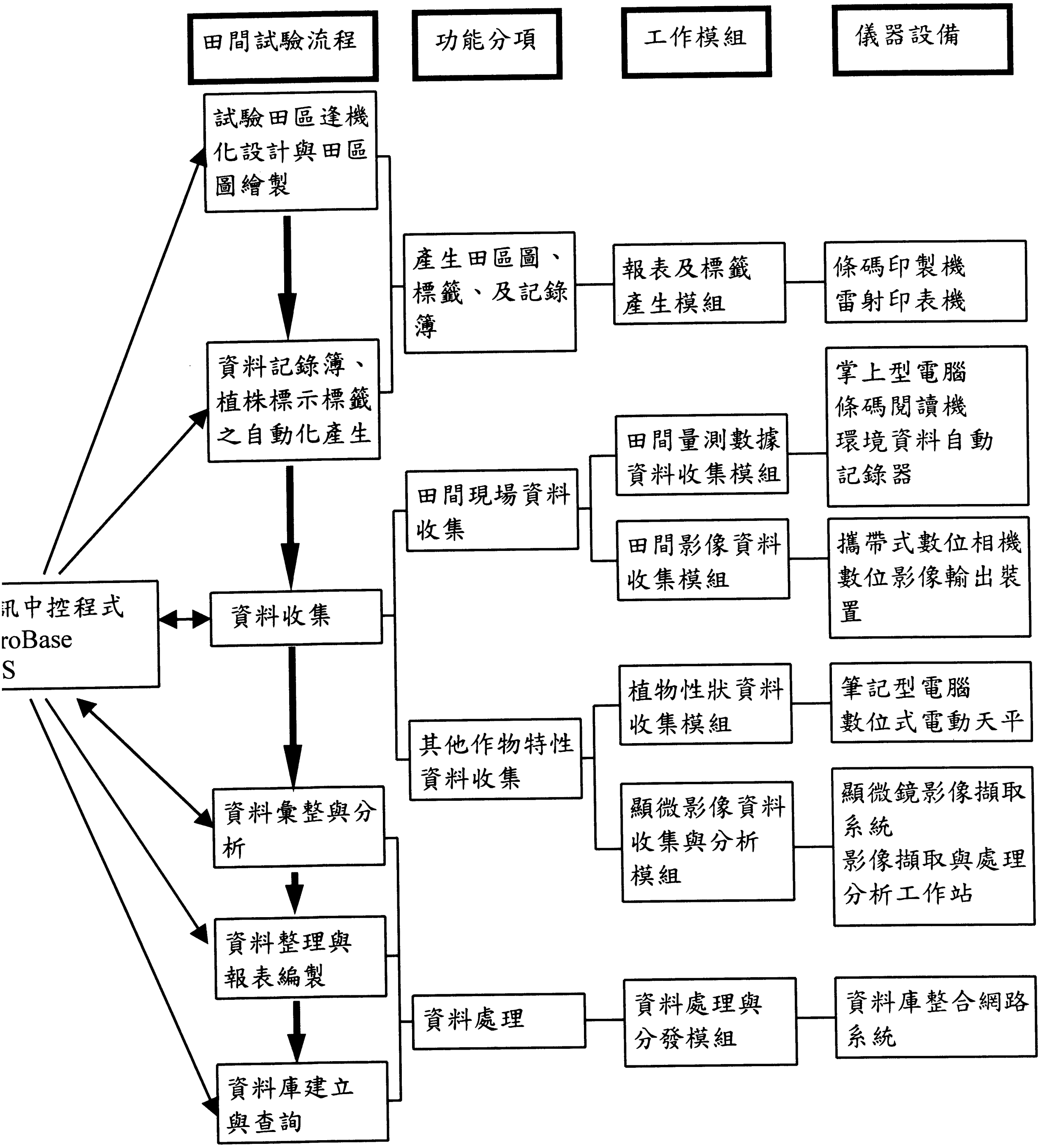
提高種原特性資料的精確性，可以藉由增加特性資料的項目與敘述的精確性來改善，如此將會大幅增加資料收集的工作量，另一方面，資料項目與敘述精確性的增加，在某個程度之後，其效益比並不一定成正比增加，因此一昧地加強這一部份，對於使用者的需求最後仍然是無法滿足。如果我們透過使用者使用種原行為的觀察，或許可以找到滿足需求的方法。使用者在取得種原材料後，大多會進行材料的繁殖與觀察，藉由實地生長的材料，觀察個別材料的外觀、株型等整體性集合性狀後，再觀察特定性狀，因此如果能夠將個別材料的外觀、株型等整體性集合性狀的影像資料包含在種原資料庫內，由使用者在挑

選材料之前，先行瀏覽影像，就減少使用者自行繁殖材料的資源而言，有絕對正面的意義。IPGRI (International Plant Genetic Resource Institute)於1998年出版的Characterization and documentation of genetic resource utilizing multimedia databases研討會專刊指出，配合資訊科技的發達，資訊工具的普及性與經濟性的改善，建議IPGRI應朝向種原多媒體資料庫方向發展。然而在收集影像資料時，需特別注意比例尺與對比及顏色的問題，以避免造成失真。

因為田間試驗之試驗單位（小區）多，調查項目亦不少，種原資料的收集是一件相當複雜的工作。將整個工作依試驗流程可劃分為試驗設計、記錄簿及標籤等文書製作、實際資料收集、彙整與分析、編製報表及建立資料庫等項目（圖一）。其中資料登錄及彙整是經常發現錯誤的工作項目：田間或種子資料自登錄至併入主要資料檔，需要經過數次傳抄的過程，最常見的錯誤是每一筆資料的品系號或重複小區序號發生錯誤，造成某些小區的資料超過一筆而另一些則無資料的情形。若能在田間或考種記錄時採用掌上型電腦輸入，或配合條碼的使用則可減少此種人為錯誤的可能。採用掌上型電腦時之考量，以易於程式控制及與桌上型電腦間之資料傳遞為主。記憶體容量及運算速度在此並不重要。此外，因為田間調查或考種時多為高溫、多濕及多塵的狀況，另一點須要考慮的則是對惡劣環境的忍受力。

直接由電腦自測量儀器取得資料的方法，依測量儀器是否具有數位訊號通訊的能力，而可分為類比訊號介面與數位訊號介面兩種資料擷取系統。一些較簡單的測量儀器（如溫度或土壤水分測定計），常是將測量之結果以相當比例的電壓變化來表示，這種訊號的變化具有連續性，稱為類比訊號，可直接接上繪圖記錄器來驅動墨筆，亦可以連接至一與電腦相連的類比—數位轉換卡上後，轉換為非連續的數位訊號，再輸入至電腦之記憶體中儲存。通常需要另外一套軟體來驅動這種類比—數位轉換卡。此外一些測量儀器已具備內建的類比—數位轉換設備，測量結果可直接在面板上顯示，如數位式的電子天平及水分測定儀等。這種儀器若配合通訊協定（通常為RS-232C，亦有用IEEE-488者）即可與個人電腦之序列埠 (Serial port)相連接，在程式控制下與個人電腦進行控制訊息與資料的交換。不論採用何種介面，最後儲存於電腦中的資料可直接轉入整體資訊網路中，而減少彙整時的錯誤。

本文將先對目前使用最多的條碼系統作初步的介紹，再以RS-232C通訊協定說明電腦與測量設備的連接方式，最後將以與一台數位天平連接的個案舉例說明控制程式的內容。


田間試驗流程
功能分項
工作模組
儀器設備
試驗田區逢機化設計與田區圖繪製
資料記錄簿、植株標示標籤之自動化產生
資料收集
資料彙整與分析
資料整理與報表編製
資料庫建立與查詢
訊中控程式
roBase
S
產生田區圖、標籤、及記錄簿
田間現場資料收集
其他作物特性資料收集
資料處理
報表及標籤產生模組
田間量測數據資料收集模組
田間影像資料收集模組
植物性狀資料收集模組
顯微影像資料收集與分析模組
資料處理與分發模組
條碼印製機
雷射印表機
掌上型電腦
條碼閱讀機
環境資料自動記錄器
攜帶式數位相機
數位影像輸出裝置
筆記型電腦
數位式電動天平
顯微鏡影像擷取系統
影像擷取與處理分析工作站
資料庫整合網路系統

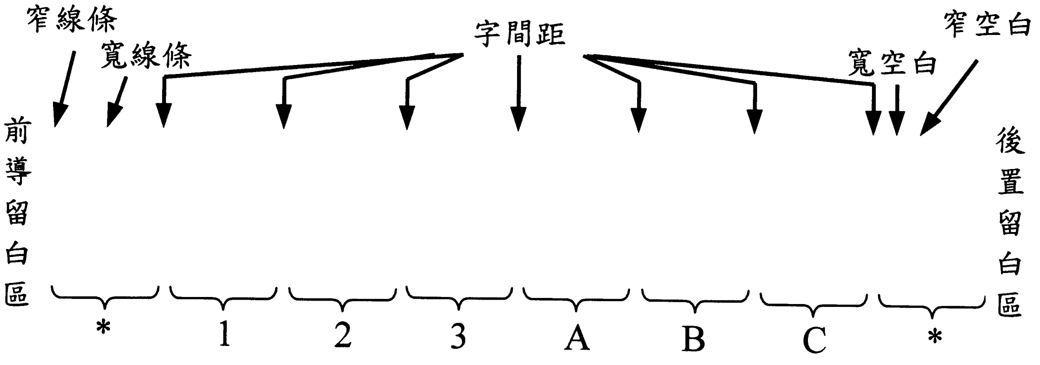
圖一、種原資訊流程解析。

二、條碼

（一）條碼的組成架構

將要輸入電腦內的所有「字元」(Character)，經過「編碼」(Coding)後，使得每一「字元」有自己相對應的「碼」(Code)。而編碼的方式是藉由許多寬度不一的“線條”(Bar)及“空白”(Space)的組合，表表示出各種不同的碼(Code)。

在一個條碼的起頭及結束的地方，都會被放入「起頭碼」(start code)及「結束碼」(end code)，用以辨視條碼的起始及結尾，並不是用來表示任何資料。每一種條碼規格的「起頭碼」及「結束碼」的圖樣(pattern)，並不完全相同。在條碼的左右兩側都必須有段空白，稱之為“留白區”(Quiet zone)。



圖二、條碼的基本結構，以39碼為例。

(二)不同編碼方式的特性

在上述的基本架構之下，有不同的編碼方式，可依下列的7種特性分別描述：

1. 字元組合 (Character Set)

「字元組合」指的是某一種條碼規格所能表示的字元 (Character) 的組合，每一種條碼規格所涵蓋的範圍及數目均不相同。有些條碼規格只能表示數字而已，如 UPC/EAN 碼；有些則能表示英文字及數字，如 39 碼；有的甚至能表示出全部 ASCII 字元表上的 128 字元，如 128 碼。

2. 符號種類 (Symbology Type)

在分散式條碼中，每一個字元可以獨自地被解碼，列印時每個字元與旁邊的字元間由「字間距」(Intercharacter Gaps) 分開，而且

每個字元固定是以“線條”做為結束。

相對地，在連續式條碼中，在字元之間不再有「字間距」的存在，每個字元都是由“線條”(bar)開始，而由“空白”(space)結束。而且在每一個字的結尾後，馬上就緊跟下一個字元的起頭。

3. 寬度數目 (Number of Element Widths)

大多數的條碼規格都使用兩種寬度；在許多條碼的使用中，‘寬’及‘窄’的比例介於 2.0～3.0 之間。這個比例值越大，對列印品質不良的容許度就越大。

使用多種寬度的條碼中“線條”及“空白”可有許多不同寬度值，而不再侷限於僅使用兩種寬度。大部份“使用多種寬度”的條碼規格所表示的字元均可以被細分成許多「區間」(Module)，而“線條”及“空白”則是「區間」的整數倍。「連續式」且使用多種寬度的條碼規格經常被稱為 (n, k) 碼，其中 ‘n’ 為在字元內的「區間」數，而 ‘k’ 則為“線條”(或“空白”)的數目。而 (n, k) 碼可表示的字元圖像 (pattern) 的數目，可由下列公式求得：

$$(2k-1)!(n-2k)!$$

最著名的 (n, k) 碼應首推 UPC/EAN 碼，它是 (7,2) 碼，所以說，在每個字元內，可細分成 7 個「區間」，用來放入 2 個“線條”和 2 個“空白”。將值代入公式後，我們可以得 20，換句話說，UPC/EAN 碼最多可表示出 20 種字元圖像的組合。

4. 固定或可變長度 (Fixed or Variable Length)

這個特性指的是，在條碼中包含的資料長度是固定或是可變的。有些條碼規格因限於本身結構的關係，只能使用固定長度的資料，UPC/EAN 碼便是一個最好的例子。

但是在某些系統雖然使用的是“可變長度”的條碼規格，可是在設計時，為了確保資料的正確性，故意限制成僅使用固定長度的資料，所以當解讀出來的資料長度不合時，電腦立即可以查知掃瞄有誤。

5. 窄線條的寬度 (X)

‘X’ 值指的是，條碼中窄“線條”及“空白”的寬度。它通常是某個條碼中所有窄的“線條”及“空白”的平均值，而且它使用的單位通常是“mil”(千分之一英吋，即 0.001 inch)。

6. 密度 (Density)

「密度」指的是，在一固定長度內可表示字元數目。所以如果我們說，條碼規格 A 的「密度」高於條碼規格 B 的「密度」，則表示當兩者使用相同的 'X' 值(窄線條的寬度)時，在同一長度內，在條碼 A 內可容納下較多的字元。

大部份「連續式」的條碼規格，由於並無「字間距」的存在，在同樣的空間內可列印較多的字元，所以「密度」會比一般的「分散式」的來得高。但相對地，連續式條碼的密度較高，它對印表機的列印精密度的要求，相對地也就比較高。

7. 自我檢查 (Self-Checking)

「自我檢查」指的是，某個條碼規格是否有自我檢測錯誤的能力，會不會因一個列印上的小缺陷，而可能使得一個字元被誤判成為另外一個字元。一個經過精心設計的條碼規格，應該會有自我檢測誤判的能力，例如常見到的 39 碼，就具備有這種功能。

沒有「自我檢查」能力的條碼規格，在使用上大多有「檢查碼」的設定，EAN/UPC 碼便是最好的例子；相對地，有「自我檢查」能力的條碼規格，則大多沒有硬性規定要使用「檢查碼」。

(三) 常用條碼規格

	使用檢查碼	使用寬度數目	(n, k)碼	分散式/連續式	可表示字串長度	可表示字元數目
39 碼	?	二	×	分散式	可變	44
UPC-A 碼	V	四	(7, 2)	連續式	12	10
UPC-E 碼	V	四	(7, 2)	連續式	6	10
EAN-13 碼	V	四	(7, 2)	連續式	13	10
EAN-8 碼	V	四	(7, 2)	連續式	8	10
交錯式 25 碼	?	二	×	連續式	偶數	10
Codebar 碼	×	十八	×	分散式	可變	16
128 碼	V	四	(11, 3)	連續式	可變	106

備註：使用檢查碼欄位中：使用(V)，不使用(×)，可用可不用(?)。

39 碼是第一個包含數字及英文字母的條碼規格。除了在超級市場、零售業的應用中使用 UPC/EAN 碼外，幾乎在其他的應用環境中，都是使用 39 碼，可以說是，目前使用最為廣泛的條碼規格，也正因為如此，支援 39 碼的軟硬體設備也最齊全。

每個39碼的字元，均是由五條“線條”及四條“空白”構成的，而且四條“空白”夾雜在五條“線條”之中，這九個線條中固定有三個寬的及六個窄的。

39 碼原先只能用來表示 43 個資料字元，但在“Full ASCII Mode”中，就利用這 43 個字元來組合出 128 個字元。當讀碼器被設定為“Full ASCII Mode”時，字元$、/、%、+則被當成「先行碼」(precedence code)，當這四個字元後面再跟著 26 個英文字母中任何一個，即被解釋成其他字元。

要傳達資料太長時，可以將全部資料分成數個較短的資料，這樣子有時會比較方便條碼的列印及讀取。整個“串連”(concatenation) 的原則是，當讀碼機讀到一個條碼內資料，假設其第一個字元是‘SPACE’的話，則表示下面還有資料在別的條碼上，此時讀碼機會先去除資料的第一個字元(即‘SPACE’)，再存入緩衝區 (buffer) 內；再繼續掃瞄下一個條碼，如此不斷地將所讀到的資料串連到緩衝區中，一直到所讀出資料的第一個字元不是‘SPACE’為止，此時讀碼機會先將資料存入緩衝區內，再將所有在緩衝區內的所有資料，當成一筆完整的資料傳送出去。

雖然39碼的密度低，在通常有限的空間中無法編入較長的訊息，但因為其字元寬度固定，編碼容易，可用「條碼字型」的概念，配合文書處理軟體的合併列印功能，直接將所要編碼的內容轉換為條碼。而其自我校正的功能加上抗污性強，使得39碼成為田間及溫室條碼標籤的最佳選擇。

UPC碼是“Universal Product Code”的簡稱，UPC 碼是美國自1973年起專門設計來在超級市場中應用，它給每個一產品一個代號所以可用來識別某一特定產品及其生產廠商。它的兩種版本中 UPC-A 碼設計成固定容納 12 個數字。UPC-E 碼只能用來編碼六個數字，所以一般用在較小包裝的產品上。

EAN 碼是“European Article Numbering System”的縮寫，它是於西元 1977 年，由歐洲十二個工業國家共同制定出來的，是改良 UPC 碼而來，這兩者間有許多相似之處。可以說 EAN 碼是 UPC 碼的「Superset」，可以解讀 EAN 碼的讀碼器一定可以解讀 UPC 碼，但是可以解讀 UPC 碼的讀碼器卻不一定可以解讀 EAN 碼。EAN編碼中，每一個國家有屬於自自己的“國家代碼”(Country Code)，台灣的國家碼為

47.1。

每一本書都有一個 ISBN 碼，方便圖書館管理。ISBN 碼是 EAN-13 碼的變形，其中「國家碼」固定為“978”。

交錯式 25 碼最大的特色在於，它的每一個字元（character）是由兩個數字（numeric digit）交錯組成的，一個使用五根“線條”來表示，另一個則使用五根“空白”來表示，它們彼此間是彼此交錯在一起的。也正因為如此，交錯式 25 碼編碼資料的長度必定為偶數，這也是「交錯」(interleave) 這個字眼的由來。

Codebar 碼是於西元1972年發展出來的，如今大多應用於圖書館、血液銀行以及小包裹快遞等應用上。在 CodeBar 碼中，所有字元均由四條“線條”及三條“空白”來代表，換句話說，三條“空白”夾雜在四條“線條”之中，所以由此我們可以推想出 CodeBar 碼是屬於「分散式」的條碼規格。

128碼是（11, 3）碼，所以它的‘n’值為 11，代表著在每個字元內，可細分成 11 個「區間」(module)；而‘k’值則為 3，則表示在每個字元內，會有 3 個“線條”和 3 個“空白”。(11, 3) 碼可用來表示 252 種不同的字元圖像（pattern），可是 128 碼只能表示 106 種字元，可見有 146 種排列組合的方式並未使用到這是為了確保資料的正確性，減少誤判的機率。

（四）讀碼機（Bar Code Reader）架構

每一種條碼規格都自有其編碼的方法，但不管如何，在一種條碼規格中，雖然有許多「碼」存在，但其中每一個碼自有其獨特的“線條”及“空白”的組合方式，與其他碼不同。這些記號必須經由被稱為「讀碼機」(Bar Code Reader) 的特殊光學儀器才能夠被解讀出來，而這整個解讀過程可以細分為下列三個動作：

1. 掃瞄：

使用光電技術來掃瞄條碼，依照條碼的線條反映回來明亮度不同，想像得到的是，條碼中的白線條（Space）會比黑線條（Bar）反映回來更多的光度，再找出光度變化較大的地方，即白線條與黑線條交接之處，而實際上傳出的信號在黑白交接之處會有較強的信號。

2. 解碼：

由傳入的電子信號找出，分析出黑、白線條的寬度，再進一步根據各式條碼的編碼原則，將條碼資料予以解析出來。

3.傳送：

將解讀出來的條碼資料傳送給電腦主機。解碼器與主機的連接可能透過 RS-232, RS-422 或 RS-485 等介面。甚至有些可直接插入鍵盤的介面孔中，模擬鍵盤的輸入。

有些讀碼機為獨立且可隨身攜帶的單元，除以將上述三項動作整合到一部單機之外，本身具有可暫存的記憶體，可於工作完成後再透過介面將資料整批輸回主電腦。若具有部分的可程式化設定能力，則可成為田間或溫室資料收集上最有效率且經濟的工具。

三、RS-232C傳輸介面

(一) 資料編碼

1.信號位準

平常信號線呈-12V電位，假如信號線在某處斷線時，位準變成懸浮狀態（電壓變成零），就得以判斷線路發生問題。

狀態	“L” (low)	“H” (high)
電壓範圍	-3~-25V	+3~+25V
邏輯	“1”	“0”
名稱	Mark	Space

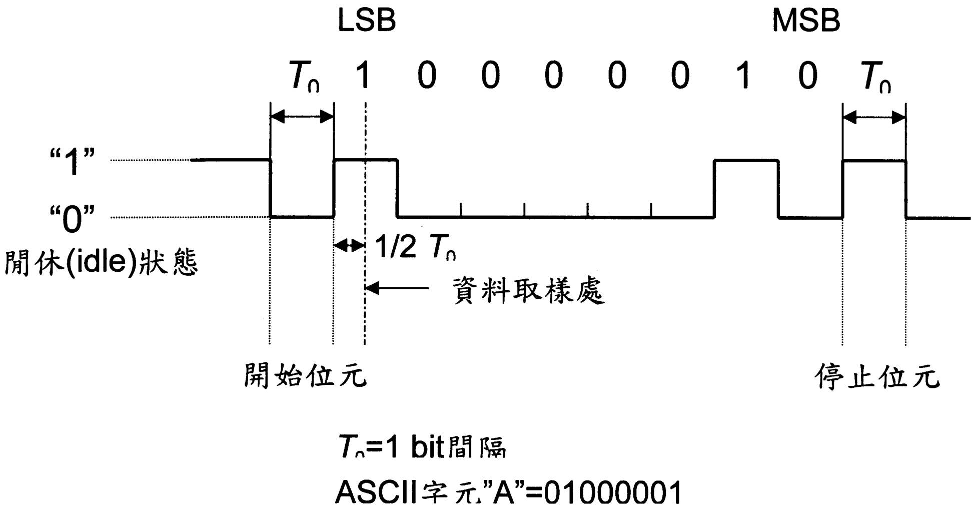
2.開始位元(Start Bit)與停止位元(Stop Bit)

當序列介面處於“閒休”(idle)狀態而不進行資料之授受時，資料線呈“1”之狀態，這樣一來，開始傳送資料時，假如最初之資料為“1”之場合，就無法區分資料與閒休狀態。因此，在開始發送資料時，為了通知接收裝置，送出1個位元之“0”信號，我們稱呼此位元為“開始位元”(start bit)，緊跟著開始位元後面連續送出的資料位元（8位元之場合共有8個資料位元）稱為“data bit”或“character bit”（字元位元）。圖三所示為傳送ASCII (American Standard Code for Information Interchange)字元“A”時之情形。

在傳送資料時，資料發送裝置首先令資料線(data line)送出開始位元（1個位元之“0”資料），然後接著由資料位元組(data byte)之最低位元(LSB)開始發送資料，各個資料位元保持“0”或“1”之狀態的時間等於T_0，T_0乃是取決於資料傳送速度的一定期間，例如，當傳送速度為

1200鮑(Buad)時，T_0=0.833ms，當傳送速度為300鮑時，則T_0=3.33ms。

在接收裝置方面，當其檢測出開始位元時，立即進入準備接收資料的狀態。接收裝置為了捕捉資料位元之各個位元的中心位置的資料，在等待1/2 T_0時間後每隔T_0期間逐一取入資料，等到完成最後一個資料(MSB)之取樣時，將資料轉換成並列資料，送往CPU。



圖三. 序列埠資料傳送信號位準編碼

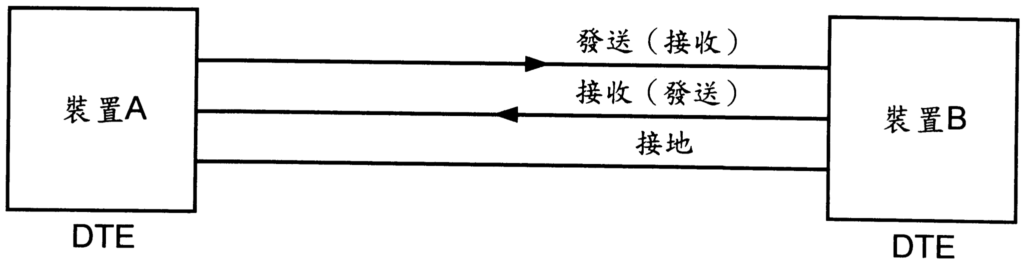
在資料發送裝置方面，當其送完最後一個資料位元時，再度令資料線回復到閒休狀態 1"，如果是接著傳送下一個資料時，則先行送出" 1"之資料（保持T_0之時間），然後再一次送出開始位元，開始發送下一個位元組(byte)的資料，我們稱呼保持T_0時間之 1"資料位元為 停止位元"(stop bit)。

這個停止位元的功能不僅是在於與下一個資料的開始位元有所區別，而且，也提供接收裝置足夠的時間來處理資料，停止位元的持續時間有1 bit、1.5 bit或2 bit等。

由以上所述得知，序列介面在傳送資料時，其傳輸動作是否正確無誤取決於時序(timing)之正確與否。假如資料發送裝置和資料接收裝置兩者之資料傳送速度不一致時，則可能導致同一個資料被讀取兩次或者是漏讀之問題發生。

由圖三所示得知，一個字元(character)內之資料位元(data bit)必需保持正確之間隔T_0，至於第一個字元的最後一個資料位元與第二個字元的第一個資料位元之間的時間間隔只要大於停止位元之時間即可，時間長短不受限制，這種傳送方式稱為“調步同步式”或“非同步式”資料傳送方式。

不過，序列介面之資料傳送並非僅使用一條信號線，而必需使用共用地線，合計二條信號線。假如是作雙方向資料傳送之場合，則必需再增加一條信號線而變成三條信號線，如圖四所示。圖中的DTE為Data Terminal Equipment 之縮寫，指的是如終端機或電腦等裝置；DCE為Data Communication Equipment之縮寫，指的是數據機之類的通訊設備。



圖四、基本的訊號傳輸線路

3.同位元檢測

同位元檢測(Parity Check)是用來檢查所收到資料的正確性。同位元檢測分為奇同位元檢測與偶同位元檢測兩種，通訊的兩端設定必須一樣。奇同位的意思就是整個資料段加上同位元之後，所有”1”的個數一定是奇數，若是偶數就表示所收到的資料（或同位元）發生錯誤，應請對方重傳。例如資料”01001101”之中共有四個”1”，若設定為奇同位時，最後還要加上一個同位元”1”，變成”010011011”，使”1”的個數成為奇數時才發送出去。若原資料為”01001111”，則同位元不再設定，發送”010011110”。

（二）流通控制

Pin No.	電路名稱	DTE-DCE 方向	慣用略號	機　能　概　要

1	安全用接地	—	FG	裝置外框體的連接
2	發送資料	→	TxD	DTE 發送資料用
3	接收資料	←	RxD	用以接收對方送來之資料
4	發送要求	→	RTS	控制 DCE 之發送狀態
5	可發送	←	CTS	表示 DCE 處於可發送之狀態
6	Data Set Ready	←	DSR	表示 DCE 處於可動作之狀態
7	信號用接地	—	GND	各相互連接電路之基準電位
8	資料載波檢出	←	DCD	表示接收信號之狀態
20	資料終端機備妥	→	DTR	表示 DTE 處於能動作之狀態
22	被召喚顯示	←	RI	表示 DCE 是否正在接受召喚(Call)信號

除了發送、接收、地線等3條線以外的信號線全部是用來控制調變解調器(MODEM)的控制線以及表示調變解調器之狀態的信號線，通稱為“控制線”。以下說明各控制線之機能：

1. DTR (Data Terminal Ready)：資料終端機備妥，pin 20

表示DTE處於動作狀態。通常，令此控制線ON時，則表示DCE要求DTE進入動作狀態；令此控制線OFF時，則表示DTE已完成傳送資料給DCE的動作，通信回線被關閉。

2. DSR (Data Set Ready)：資料發送接收裝置備妥，pin 6

表示DCE處於動作狀態。這一條控制線受DTR之控制而被啟動，當DCE打開通信回線後這條控制線變成ON。

3. RTS (Request to Send)：要求輸送資料，pin 4

DTE送出RTS信號要求DCE輸送資料給DTE，當此控制線處於“ON”狀態時，則DCE將載波輸送到通信回線上。在DCE輸送資料給DTE之期間內，RTS線必需保持“ON”之狀態。

4. CTS (Clear To Send)：pin 5

這條控制線用以表示DCE處於能夠發送資料之狀態。RTS信號啟動這條信號線，等待送往通信回線之載波穩定後變成ON狀態。

5. DCD (Data Carrier Detect)：資料載波檢出，pin 8

當DCE經由通信回線檢測到資料接收所需之載波時，這條控制線變成“ON”狀態。利用這條控制線，可以確認對方之DCE是否能夠發送資料過來。

6. RI (Ring Indicator)：振鈴指示，pin 22

當DCE接受到來自通信回線的呼叫信號時，這條控制線立即變成"ON"狀態。所謂呼叫信號，就是相當於使用公眾回線之傳真機或內含調變解調器之電話機在接收到來自外線的振鈴信號（鈴聲響時）。

傳送裝置大多使用8251作為UART(Universal Asynchronous Receiver Transmitter)，然而，8251並無與DCD相當的功能，因此，個人電腦之間的資料通信幾乎全部不使用DCD，在連接2部終端機(terminal)時，只要將DTR和DSR、RTS和CTS、TxD和RxD成對地連接在一起就算大功告成了。

有些資料傳送裝置並未具備上述的所有機能，在此場合，可採用迴環(loop back)方式或者是短接線(Jumper)方式將多餘功能之端子拉昇(pull up)至+12V。

(三) 數位天平的連接與控制

以Precisa 2200C機型電動數位天平為例，說明其與電腦的連接線路、指令及輸出格式、以及一個簡單的控制程式。

1. 連接線路

因為數位天平是一個能接受指令與產生資料的設備，在RS-232C的通訊協定中是被分類為與電腦相同的DTE，在線路的連接上，採用「2-3跳線」的方式。也就是把電腦的第二針腳（發送）接到天平的第三針腳（接收），而把電腦的第三針腳（接收）接到天平的第二針腳（發送）上，這種跳線的接線方式也稱為「虛擬數據機」(Null Modem)，因為原先在兩部DTE之間所配置的兩部DCE的功能，由這一條電纜所取代了。

本例所用的數位天平與電腦之間的資料交換量低，速度也不高，因此不需要流通控制，只需要再連接一條共用的信號接地線（兩邊的第七針腳互連），總共三條線就可以了，資料交換的控制由軟體來維持。

2. 通訊協定

本例所用的數位天平，依其使用手冊，通訊的參數設定為9600 Baud, 7 data bit, one stop bit, Even parity。

天平指令種類與格式

常用且與本例有關指令說明如下：

指令	功能
T	Tare

T-100	Set tare value to -100g
S0	Stability low
S1	Stability middle
S2	Stability high
P0	Send single value
P1	Send single stable value
P2	Send single stable value automatically after load change
P3	Send value continuously after every integration time

每次透過連接介面對天平下指令時，在指令之後必須緊接著包含<CR>與<LF>兩個控制字元。

天平資料輸出格式

電動天平每次輸出都是固定長度的20個字元，依序為前導的4個空白字元(blank)、一個正負號字元、7個數字字元（其中包括一個小數點）、2個空白字元、三個代表單位的英文字母字元及<CR>與，<LF>，舉例如下：

				+	1	0	4	.	5	6	7	8					g	CR	LF
				+		6	7	.	2	3	8	7			d	w	t	CR	LF
				+		3	.	3	6	1	9	3			o	z	t	CR	LF
				+				4	8	2	1	6			p	c	s	CR	LF

5. 控制程式

```
10   OPEN "COM2:9600,E,7,1,CD0,CS0,DS0,OP0,LF,RS" FOR RANDOM AS #1
20   FILE$="BALANCE.LST"
30   OPEN FILE$ FOR APPEND AS #2
40   FRAMECHK$=STRING$(6, 32)+STRING$(1, 13)+STRING$(1, 10)
50   PRINT #1, "T"
60   PRINT #1, "S0"
70   PRINT #1, "P2"
80   INPUT "INPUT ENTRY NUMBER OR END:", ENTRY$
90   IF ENTRY$="END" THEN GOTO 240
100  ANSWER$=INPUT$(20, 1)
110  FRAME$=LEFT$(ANSWER$, 4)+MID$(ANSWER$, 14, 2)+
     RIGHT$(ANSWER$,2)
120  IF FRAME$=FRAMCHK$ THEN
130  WEIGHT$=LEFT$(ANSWER$, 18)
140  ELSE WEIGHT$="COMMUNICATION ERROR"
150  END IF
160  PRINT "ENTRY NO; WEIGHT", ENTRY$, WEIGHT$
170  PRINT #2, "ENTRY NO; WEIGHT", ENTRY$, WEIGHT$
180  GOTO 80
240  CLOSE
```

圖五、控制數位天平的BASIC程式

6. 程式說明：

行號	內容
10	開啟第二通訊埠作為可隨機存取的#1 檔案。
20	定義文字變數 FILE$（檔案名稱）的內容為 WANG.LST。
30	開啟一個名字為 FILE$（WANG.LST）的檔案，並稱呼它為#2 檔案。如果該檔案已經存在，則新資料附寫於檔案的最後面。
40	定義一個用來檢查資料傳輸格式的特殊文字變數 FRAMECHK$，其內含包括 6 個空格，一個 CR 及一個 LF 字元。
50	向#2 檔案（第二通訊埠）送出字串"T"，指示電動天平歸零。
60	向#2 送出字串"S0"，指示電動天平採用較低的穩定度標準。
70	向#2 送出字串"P2"，指示電動天平在承載的重量改變並達到平衡後送出測量值。
80	輸入小區編號並存放於字串 ENTRY$中。
90	檢查 ENTRY$字串的內容，如果是"END"，就跳到 240 行（結束）。
100	從#1 檔案輸入 20 個連續字元，結果存到文字變數 ANSWER$中。
110	取 ANSWER$中最左的四個字元，加上第 14 個字元起兩個字元，在加上最右邊的兩個字元，組成一個新的變數 FRAME$。

120	如果 FRAME$的內容與先前定義的 FRAMECHK$相同，表示資料傳輸無誤，就進行下一行。
130	把讀到結果的前 18 位字元寫入 WEIGHT$文字變數中。
140	如果 FRAME$的內容與先前定義的 FRAMECHK$不同，表示資料傳輸錯誤，將警告訊息寫入 WEIGHT$變數中。
150	結束 IF 控制。
160	將小區編號及測得重量顯示於螢幕上。
170	將小區編號及測得重量寫入#2（WANG.LST）檔案的最後。
180	跳回 80 行，繼續下一個樣品的輸入。
240	關閉所有已經開啟的檔案，準備結束程式。